# Phân hạng gen gây bệnh sử dụng học tăng cường kết hợp với xác suất tiền nghiệm

## Disease Gene Prioritization using Reinforcement Learning with Priors

**Đặng Vũ Tùng, Dương Anh Trà, Lê Đức Hậu, Từ Minh Phương**

***Abstract**: Disease gene prioritization is the process of ranking candidate genes according to their relevance to a disease phenotype, thus facilitating the identification of disease genes by narrowing down the set of genes to be tested experimentally. Many methods have been proposed for disease gene prioritization based on relationships between proteins encoded in protein-protein interaction networks using various graph-based algorithms. In this paper, we propose a novel method for prioritizing candidate disease genes by combining reinforcement learning with PageRank algorithm and assigning priors for known disease genes. We experimentally evaluate the proposed method on a human protein interaction network and compared its performance with a state-of-the-art methods, namely PageRank with priors, Random Walk with Restart and K-Step Markov. The experiment results show that our method achieves relatively high performance in terms of AUC values and outperforms comparative methods.*

### I. MỞ ĐẦU

Xác định gen gây bệnh là bài toán quan trọng trong y sinh học và sinh học phân tử. Trước đây, việc xác định gen gây bệnh được thực hiện chủ yếu bằng các thực nghiệm sinh học. Phương pháp này được thực hiện cho hàng trăm gen ứng viên nằm trên một vùng nhiễm sắc thể khả nghi nên đòi hỏi nhiều thời gian và chi phí rất cao. Phân hạng gen là sử dụng các phương pháp tính toán để sắp xếp các gen ứng viên sao cho các gen có khả năng liên quan tới bệnh được nhận thứ hạng cao hơn. Sau khi phân hạng, một nhóm nhỏ các gen với thứ hạng cao sau đó sẽ được lựa chọn để kiểm nghiệm bằng thực nghiệm.

Khái niệm về phân hạng gen được giới thiệu lần đầu tiên vào năm 2002 bởi Perez-Iratxeta và cộng sự [1]. Trong bài báo, Perez-Iratxeta và cộng sự đã mô tả phương pháp tiếp cận tính toán đầu tiên để giải quyết vấn đề này. Kể từ đó, nhiều phương pháp tính toán khác nhau sử dụng các chiến lược, các thuật toán và nguồn dữ liệu khác nhau đã được phát triển [2,3]. Trong các phương pháp phân hạng đã có, phương pháp dựa trên mạng sinh học như mạng tương tác protein được sử dụng khá phổ biến do dữ liệu tương tác protein/gen ngày càng đầy đủ và đa dạng [3]. Các phương pháp này dựa trên một quan sát đó là các gen liên quan đến cùng một bệnh hoặc các bệnh tương tự có xu hướng nằm gần nhau trong mạng tương tác gen/protein (còn được gọi là đặc tính "mô đun bệnh"). Để có thể sử dụng kỹ thuật phân hạng này, ngoài dữ liệu mạng sinh học, cần có thuật toán phân tích mạng/đồ thị và xếp hạng các nút trên đồ thị. Do các mạng sinh học trên thực tế có các đặc tính cấu trúc tương đồng với các mạng xã hội và mạng web như "kích thước tự do" (scale-free) và "thế giới nhỏ" (small-world) [4], nhiều nghiên cứu đã áp dụng các thuật toán được sử dụng để phân hạng các nút trong mạng xã hội và mạng web để phân hạng các gen/protein trong các mạng sinh học [5].

Các phương pháp phân hạng gen gây bệnh dựa trên mạng tương tác protein được phân thành phương pháp cục bộ và phương pháp tổng thể [2]. Phương pháp cục bộ chỉ xem xét các gen lân cận với gen gây bệnh đã biết như các gen được kết nối trực tiếp hoặc thông qua đường đi ngắn nhất. Các phương pháp tổng thể sử

dụng các thuật toán lan truyền thông tin bệnh từ các gen gây bệnh đã biết qua hệ thống mạng để gán cho các gen ứng viên các điểm số đánh giá mức độ tương đồng với các gen gây bệnh đã biết, cũng được xem là mức độ liên quan với bệnh đang được xem xét.

Ví dụ, phương pháp bước ngẫu nhiên có quay lại (RWR: Random Walk with Restart) [6] khai thác cấu trúc tổng thể của mạng dựa trên hành vi của một chuyển động ngẫu nhiên trên một mạng hay đồ thị. Theo hành vi này, một thực thể xuất phát từ một nút khởi đầu sau đó di chuyển trên đồ thị bằng cách chuyển đến các nút lân cận một cách ngẫu nhiên với xác suất tỷ lệ với trọng số của các cạnh kết nối. Tập hợp các nút trong quá trình di chuyển là một chuỗi Markov và được gọi là một bước ngẫu nhiên trên đồ thị (random walk on graph). Tại thời điểm bất kỳ trong quá trình di chuyển, thực thể cũng có thể quay lại nút khởi đầu với một xác suất nhất định được gọi là xác suất quay lại (back-probability). Khi đó chúng ta có thể coi đây là bài toán bước ngẫu nhiên với các xác suất tiền nghiệm (random walk with priors). Các nút được thăm nhiều hơn được coi là có độ quan trọng lớn hơn. Đại lượng này đánh giá tầm quan trọng tương đối/độ tương tự của các nút còn lại so với tập các nút gốc.

Ưu điểm chính của phương pháp bước ngẫu nhiên là tốc độ thực hiện nhanh do đó có thể áp dụng cho các mạng có kích thước lớn. Khi áp dụng thuật toán này cho bài toán phân hạng gen gây bệnh, các gen gây bệnh đã biết đóng vai trò như các nút khởi đầu, các gen còn lại trên mạng được xem là các ứng viên. Kohler và cộng sự [7] đã áp dụng thuật toán này trên các mạng tương tác protein để xác định các gen gây bệnh mới. Kết quả thử nghiệm trên một tập gồm 110 nhóm bệnh cho thấy phương pháp này đạt được hiệu năng dự đoán tốt.

Duc-Hau Le và cộng sự [8,9] đã cải tiến phương pháp RWR bằng cách tăng cường trọng số hàng xóm của các gen gây bệnh đã biết. Cũng xuất phát từ ý tưởng sử dụng các xác suất tiền nghiệm, Chen và cộng sự [10] đã sử dụng các thuật toán phổ biến trong phân tích mạng xã hội và mạng Web dùng để đánh giá tầm quan trọng tương đối của nút như: HITS with priors, PageRank with priors và K-step Markov cho bài toán phân hạng các gen ứng viên trên các mạng tương tác protein.

Một cách tiếp cận khác sử dụng xác suất tiền nghiệm là PRINCE (PRIoritizatioN and Complex Elucidation) được phát triển bởi Vanunu và cộng sự [11]. PRINCE sử dụng thuật toán lan truyền để dự đoán gen bệnh dựa vào thông tin tích hợp giữa kiểu hình bệnh và mạng tương tác protein. Phương pháp này tính toán mối liên quan giữa một bệnh và gen bệnh đã biết với một bệnh khác sử dụng hàm logistic dựa trên sự tương tự kiểu hình giữa hai bệnh. Gen liên quan tới bệnh sau đó được sử dụng như xác suất tiền nghiệm để xây dựng chức năng phân hạng gen [3].

Trong bài báo này, chúng tôi sử dụng một phương pháp phân hạng mới RL_Rank được đề xuất bởi Derhami và cộng sự [12,13] dựa trên sự liên kết của các nút trong đồ thị và khái niệm về Học tăng cường để phân hạng các trang Web. Xuất phát từ sự thành công của các thuật toán trên trong việc sử dụng "thứ hạng đầu" hay xác suất tiền nghiệm, để biến độ quan trọng tuyệt đối của các nút trong mạng thành độ quan trọng tương đối/độ tương tự của các nút đối với một tập các nút gốc, chúng tôi cải tiến thuật toán RL_Rank thành thuật toán RL_Rank with priors bằng cách bổ sung thêm các xác suất tiền nghiệm nhằm mục đích nâng cao hiệu quả của thuật toán. Thuật toán này được cài đặt và thử nghiệm cho bài toán phân hạng và tìm kiếm gen gây bệnh dựa trên bộ dữ liệu mạng tương tác protein. Kết quả thực nghiệm cho thấy độ chính xác của phương pháp đề xuất tốt hơn so với phương pháp PageRank with priors trên cùng bộ dữ liệu thử nghiệm.

Các phần còn lại của bài báo được bố cục như sau: Phần II là phát biểu bài toán và các nghiên cứu liên quan. Phần III trình bày các kết quả thực nghiệm. Cuối cùng là phần kết luận nêu các đóng góp chính của bài báo và đề xuất các hướng cải tiến mới.

## II. CÁC NGHIÊN CỨU LIÊN QUAN VÀ GIẢI PHÁP PHÂN HẠNG GEN ĐỀ XUẤT

Trong phần này, đầu tiên chúng tôi tóm tắt một số khái niệm và phương pháp liên quan, cần thiết cho việc trình bày phương pháp đề xuất. Sau đó, chúng tôi mô tả nguồn dữ liệu sử dụng cho thực nghiệm và phương pháp đánh giá kết quả thực nghiệm.

### II.1. Bài toán phân hạng nút trên đồ thị

Mạng tương tác protein trong bài báo này được biểu diễn bởi một đồ thị vô hướng có trọng số $G = (V, E)$ trong đó tập các nút $V$ là các gen/protein và tập các cạnh $E$ thể hiện tương tác giữa các gen/protein. Giả sử cho biết trước $S$ là tập các gen bệnh đã biết (còn gọi là tập hạt giống hay tập nút gốc), tức là một số lượng nhỏ các gen đã được phát hiện có liên quan đến bệnh trong các nghiên cứu trước đó. Bài toán phân hạng gen gây bệnh được định nghĩa như sau: Cho $G$ và tập các nút gốc $S$ ($S \subseteq V$), $T$ ($T \subseteq V$) là tập bao gồm S và các nút có liên kết với các nút trong S. Hãy phân hạng tất cả các nút trong $T$ theo độ liên quan với $S$. Độ liên quan của một nút $t \in T$ được định nghĩa là trung bình cộng độ liên quan của $t$ với các nút trong $S$.

$$I(t|S) = \frac{1}{|S|}\sum_{r\in S} I(t|r) \quad (1)$$

### II.2. Thuật toán PageRank with priors

PageRank with priors là sự mở rộng của thuật toán PageRank truyền thống để tạo ra thuật toán phân hạng tùy biến [14] [15]. PageRank with priors cho phép phân hạng các nút trên đồ thị trong mối tương quan với một tập các nút gốc cho trước.

Theo PageRank, thứ hạng của một nút $v$ được tính theo công thức:

$$PR(v) = \frac{1-d}{N} + d\sum_{u=1}^{d_{in}(v)} \frac{PR(u)}{d_{out}(u)} \quad (2)$$

trong đó: $N$ là tổng số các nút, $d$ ($0 < d < 1$) là hệ số suy giảm và thường được chọn là 0.85, $d_{in}(v)$ là bậc vào của nút $v$, $d_{out}(u)$ là bậc ra của nút $u$. Ý nghĩa của PageRank là thứ hạng (độ quan trọng) của một nút phụ thuộc vào số nút trỏ tới nút đó và thứ hạng của những nút này. Hai giá trị này càng lớn thì thứ hạng của nút đang xét cũng càng lớn.

Ý tưởng của PageRank with priors là định nghĩa một vector $p_S = \{p_1, ..., p_{|v|}\}$ có xác suất trước sao cho

$$\sum_{i=1}^{|v|} p_i = 1 \quad (3)$$

và $p_v$ biểu thị độ quan trọng tương đối (hay "độ lệch ban đầu") được gán cho nút $v$. Ở đây:

$$p_v = \begin{cases} \frac{1}{|S|} & v \in S \\ 0 & v \notin S \end{cases} \quad (4)$$

Ngoài ra, PageRank with Priors cũng định nghĩa một "xác suất quay lại" $\beta$ ($0 \leq \beta \leq 1$) là xác suất quay trở lại các nút gốc trong $S$ và

$$p(v|u) = \frac{1}{d_{out}(u)} \quad (5)$$

là xác suất chuyển tới nút $v$ từ nút $u$.

Tích hợp công thức (3), (4) và (5) vào công thức (2), chúng ta thu được công thức (6) là xác suất dừng lặp (điểm phân hạng) có dạng:

$$PR(v)^{(i+1)} = (1-\beta)\left(\sum_{u=1}^{d_{in}(v)} p(v|u)PR^{(i)}(u)\right) + \beta p_v \quad (6)$$

Độ liên quan của nút $v$ tương quan với $S$ sẽ được xác định tính theo công thức $I(v|S) = PR(v)$ sau khi hội tụ.

### II.3. Thuật toán RL_Rank

Thuật toán RL_Rank sử dụng cấu trúc liên kết của các trang web và định nghĩa sự phân hạng theo hình thái của bài toán học tăng cường. Trong giải thuật này, một tác tử (agent) được xem như một người dùng lướt Web và mỗi trang Web là một trạng thái (state). Tại mỗi trang, người dùng nhắp vào một trong những liên kết có trong trang với một xác suất đều và qua đó chuyển qua trang kế tiếp. Nói cách khác, khi người dùng chọn một trang kế tiếp bằng cách nhắp ngẫu nhiên vào một trong những liên kết có trên trang hiện tại theo chính sách học $\pi$(policy) thì xác suất lựa chọn bằng $1/d_{out}$*(trang hiện tại)* với $d_{out}$*(trang hiện tại)* là bậc ra của trang hiện tại [*13*]. Khoản thưởng (reward)

dành được khi chuyển từ trang hiện tại *u* sang trang mới *v* được định nghĩa bởi công thức:

$$r_{uv} = \frac{1}{d_{out}(u)} \quad (7)$$

Điểm của trang *v* là giá trị được mong đợi của tổng các khoản thưởng đã giảm trừ mà một tác tử tích lũy được trong suốt quá trình duyệt qua các trang cho tới trang *v*. Tiếp theo, tác tử sẽ thêm khoản thưởng đã nhận được $r_{uv}$ vào các khoản thưởng tích lũy đã giảm trừ của mình. Do đó, điểm của một trang *v* là xác suất duyệt tới nó từ các trang khác được tăng lên nhiều lần bởi tổng các khoản thưởng khi chuyển đổi và các khoản thưởng tích lũy đã giảm trừ và được tính theo công thức:

$$R_{t+1}(v) = \sum_{u=1}^{d_{in}(v)} ((prob(u)/d_{out}(u)) \times (r_{uv} + \gamma R_t(u))) \quad (8)$$

trong đó $R_{t+1}(v)$ là thứ hạng của trang *v* tại thời điểm *t*+1, $R_t(u)$ là thứ hạng của trang *u* tại thời điểm *t*, $d_{in}(v)$ bậc vào của trang *v*, $prob(u)$ là xác suất về sự hiện diện của tác tử tại trang *u*, $d_{out}(u)$ là bậc ra của trang *u*, $r_{uv}$ là khoản thưởng dành cho việc chuyển từ trang *u* sang trang *v*, $\gamma$ ($0 < \gamma < 1$) là hệ số giảm trừ.

Giá trị của biểu thức $prob(u)/d_{out}(u)$ là xác suất của việc duyệt tới trang *v* từ trang *u*. Nó bằng với xác suất xuất hiện của tác tử tại trang *u* nhân với xác suất lựa chọn của trang *v* khi tác tử đang ở trạng thái *u*. Do tác tử lựa chọn một liên kết ngẫu nhiên theo phân phối xác suất đều, nên xác suất lựa chọn trang *v* từ trang *u* bằng $1/ d_{out}(u)$. Xác suất xuất hiện của tác tử tại trang thái *u* chính là thứ hạng của trang *u* trong khái niệm về PageRank, do đó $prob(u)$ được tính bằng công thức phân hạng của PageRank đối với trang *u* và $R(u)$ là thứ hạng của trang *u* thể hiện các khoản thưởng tích lũy đã giảm trừ mà tác tử nhận được cho đến khi duyệt tới trang *u*. Do đó, thứ hạng của trang *v* phụ thuộc vào bậc ra và thứ hạng của các trang có liên kết tới *v*. Kết quả thu được sẽ là một vector RL_Rank với các thành phần là điểm số/thứ hạng của các trang. Thuật toán được thể hiện trên Hình 1.

**Input:**
*V:* *Tập hợp tất cả các trang web (nút)*
*prob:* *Xác suất xuất hiện của agent tại trang u*
*R:* *Vector thứ hạng theo RL_Rank*
*ε :* *Số thực dương rất nhỏ*
**Output:**
*Vector R chứa hạng của các trang web*
**Initialize** *R, prob* vectors *//Khởi tạo ngẫu nhiên R và prob*
$\delta \leftarrow 0$
*// Tính toán các giá trị của vector prob, đây cũng chính là thứ hạng của các nút theo PageRank.*
**Do** {
**For** every page $v \in V$
$prob_{new}(v) = \frac{1-d}{N} + d \sum_{u=1}^{d_{in}(v)} \frac{prob(u)}{d_{out}(u)}$
**End for**
$\delta \leftarrow \|prob_{new} - prob\|$
$prob \leftarrow prob_{new}$
}
**While** ($\delta > \varepsilon$)
*// Sử dụng các khái niệm về Học tăng cường để tăng cường điểm cho các thứ hạng gốc của các nút để nhận được thứ hạng cuối cùng của chúng.*
$\delta \leftarrow 0$
**Do** {
**For** every page $v \in V$
$r_{uv} = 1/d_{out}(u)$
$R_{new}(v) = \sum_{u=1}^{d_{in}(v)}((prob(u)/d_{out}(u)) \times (r_{uv} + \gamma Ru)$
**End for**
$\delta \leftarrow \|R_{new} - R\|$
$R \leftarrow R_{new}$
}
**While** ($\delta > \varepsilon$)

*Hình 1. Thuật toán RL_Rank*

### II.4. Thuật toán RL-Rank with priors

Thuật toán RL_Rank cho phép xếp hạng các nút trên mạng một cách toàn cục, tức là thuật toán này tính toán độ quan trọng nói chung hay độ quan trọng tuyệt đối của các nút. Trong các bài toán tìm kiếm trên Web, cách xếp hạng này là phù hợp. Tuy nhiên, mục

tiêu của bài toán phân hạng gen gây bệnh không phải là tính độ quan trọng tuyệt đối của các nút mà là tính độ quan trọng tương đối của các nút so với các nút gốc (tức là nút tương ứng với các gen đã biết trước là gen gây bệnh). Để thực hiện việc xếp hạng tương đối như vậy, thuật toán RL_Rank cần được cải tiến phù hợp.

Để giải quyết vấn đề này, chúng tôi sử dụng ý tưởng về "thứ hạng ban đầu" hay xác suất tiền nghiệm priors trong phương pháp PageRank with priors. Trong phương pháp của mình, chúng tôi ký hiệu $S$ là tập gen gốc và định nghĩa một vector thứ hạng ban đầu $p_S = \{p_1, \dots, p_{|V|}\}$ có tổng bằng 1, trong đó $p_v$ biểu thị độ quan trọng tương đối của nút $v$. Ở đây $p_v = 1/|S|$, đối với $v \in S$ và $p_v = 0$ đối với $v \notin S$. Đồng thời, định nghĩa một xác suất quay trở lại gốc $\beta$ ($0 \le \beta \le 1$) biểu thị cho xác suất quay trở lại các nút gốc trong quá trình duyệt nhằm mục đích xem các nút gốc là quan trọng nhất. Khi đó công thức của RL_Rank được viết lại như sau:

$$R_{t+1}(v) = (1-\beta)\left(\sum_{u=1}^{d_{in}(v)}((prob_{t+1}(u)/d_{out}(u)) \times (ruv+\gamma Rtu)+\beta pv \quad (9)$$

Thuật toán tương ứng cũng được sửa lại với việc tính đến các xác suất đầu $prob(u)$ là xác suất xuất hiện của agent tại nút $u$ (theo PageRank with priors). Thuật toán được thể hiện trên Hình 2.

**Độ phức tạp của thuật toán:** thuật toán đề xuất được chia thành ba bước chính nối tiếp nhau, do đó độ phức tạp tính toán của nó sẽ bằng độ phức tạp tính toán lớn nhất trong ba bước.

Trong bước đầu tiên chúng tôi sử dụng thuật toán duyệt đồ thị theo chiều rộng (Breadth First Search), nên độ phức tạp tính toán là:

$$O_1 = O(|V| + |E|) \quad (10)$$

trong đó: $|V|$ là tổng số nút và $|E|$ là tổng số cạnh của đồ thị.

Bước tiếp theo, chúng tôi sử dụng thuật toán PageRank with priors để tìm xác suất một nút được duyệt trên đồ thị từ tập nút gốc (trong trường hợp xấu nhất, tập nút gốc trùng với tập $V$). Độ phức tạp tính toán của một lần lặp là:

$$O_2 = O(|V| + |E|) \quad (11)$$

**Input:**
*V:* *Tập hợp tất cả các gen (nút)*
*S:* *Tập các gen gốc*
*prob:* *Xác suất của agent khi duyệt tới gen j*
*priors: Vector thứ hạng ban đầu của tập tất cả các gen*
*R: Vector thứ hạng các gen theo RL_Rank*
*ε : Số thực dương rất nhỏ*

**Output:**
*Vector R chứa hạng của các gen*

*/* Sử dụng thuật toán tìm kiếm theo chiều rộng để lấy toàn bộ các gen liên kết với tập gen gốc */*
**Build** Set *T / Chứa các gen gốc và gen liên kết với chúng*
**Initialize** *R, prob* vectors *// Khởi tạo R và prob*
**Initialize** *priors /* Khởi tạo giá trị ban đầu "priors" theo công thức (4) */*
$\delta \leftarrow 0$
*// Tính toán các giá trị của vector prob, đây cũng chính là thứ hạng của các gen theo PageRank with priors.*
**Do** {
**For** every gene $v \in A$
$prob_{new}(v) = (1-\beta)\left(\sum_{u=1}^{d_{in}(v)}(prob(u)/d_{out}(u))\right) + \beta p_v$
**End for**
$\delta \leftarrow \|prob_{new} - prob\|$
$prob \leftarrow prob_{new}$
}
**While** ($\delta > \varepsilon$)
*// Sử dụng các khái niệm về Học tăng cường để tăng cường điểm cho các thứ hạng gốc của các gen để nhận được thứ hạng cuối cùng của chúng.*
$\delta \leftarrow 0$
**Do** {
**For** every gene $v \in V$
$r_{uv} = 1/d_{out}(u)$
$R_{new}(v) = (1-\beta)\left(\sum_{u=1}^{d_{in}(v)}((prob(u)/d_{out}(u)) \times (r_{uv} + \gamma Ru)+\beta pv\right.$
**End for**
$\delta \leftarrow \|R_{new} - R\|$
$R \leftarrow R_{new}$
}
**While** ($\delta > \varepsilon$)

*Hình 2. Thuật toán cải tiến RL_Rank with priors*

Giả sử sau $k$ lần lặp, thuật toán sẽ hội tụ và kết thúc, như vậy độ phức tạp tính toán tổng cộng là $k.O_2$. Do $k$ là hằng số nên độ phức tạp tính toán cuối cùng là $O_2$.

Ở bước cuối, để tăng cường điểm cho từng nút theo thuật toán Học tăng cường, chúng tôi lặp lại cách thức duyệt đồ thị tương tự như đối với thuật toán PageRank with priors, nên độ phức tạp tính toán vẫn là:

$$O_3 = O(|V| + |E|) \quad (12)$$

Như vậy, độ phức tạp tính toán của thuật toán RL_Rank with priors là tuyến tính và bằng:

$$O = O(|V| + |E|) \quad (13)$$

### II.5. Cơ sở dữ liệu mạng tương tác protein và mối quan hệ bệnh/gen

Bộ dữ liệu sử dụng cho thử nghiệm của chúng tôi được tham khảo từ [*8*] bao gồm: mạng tương tác protein mô tả bởi đồ thị vô hướng có trọng số gồm 11.886 gen và 111.943 liên kết. Dữ liệu về quan hệ bệnh-gen gồm 3284 bệnh, trong đó mỗi bệnh có từ 1 đến 31 gen liên quan đã được phát hiện. Với mỗi bệnh, tập các gen đã biết được sử dụng như là tập gốc trong quá trình phân hạng. Do sử dụng phương pháp kiểm tra chéo bỏ ra một (LOOCV: leave-one-out cross validation) để đánh giá hiệu năng của các phương pháp phân hạng, nên mỗi bệnh phải có ít nhất hai gen liên quan. Do đó từ cơ cở dữ liệu mạng tương tác protein và quan hệ bệnh-gen, chúng tôi lọc ra được 398 bệnh gây ra bởi từ hai gen trở lên và các gen này có trong mạng tương tác protein.

### II.6. Phương pháp đánh giá

Để đánh giá hiệu suất của phương pháp phân hạng, đối với mỗi bệnh chúng tôi sử dụng phương pháp LOOCV. Theo đó, với mỗi lần lặp, một gen bệnh đã biết được lấy ra và coi như là một gen ứng viên bình thường, các gen còn lại được sử dụng như các gen gốc priors làm dữ liệu đầu vào cho thuật toán. Hiệu suất của phương pháp phân hạng được xác định bằng cách tính toán giá trị AUC (diện tích dưới đường cong ROC). Cụ thể như sau: với tập gen bệnh đã biết $S$ và tập gen ứng viên $C$ (là tất cả các gen còn lại trên mạng), một gen $s \in S$ được lấy ra và chúng tôi tiến hành phân hạng tập gen $C \cup \{s\}$ theo thuật toán đề xuất với tập $S \backslash \{s\}$ được sử dụng như tập các nút gốc. Quá trình này được lặp lại cho tất cả các gen bệnh đã biết. Sau đó chúng tôi thay đổi ngưỡng $\tau$ từ 1 cho đến $|C \cup \{s\}|$. Giá trị *sensitivity* và *1-specificity* được tính toán theo các công thức:

$$sensitivity = \frac{TP}{TP+FN} \quad (14)$$

$$1 - specificity = \frac{FP}{FP+TN} \quad (15)$$

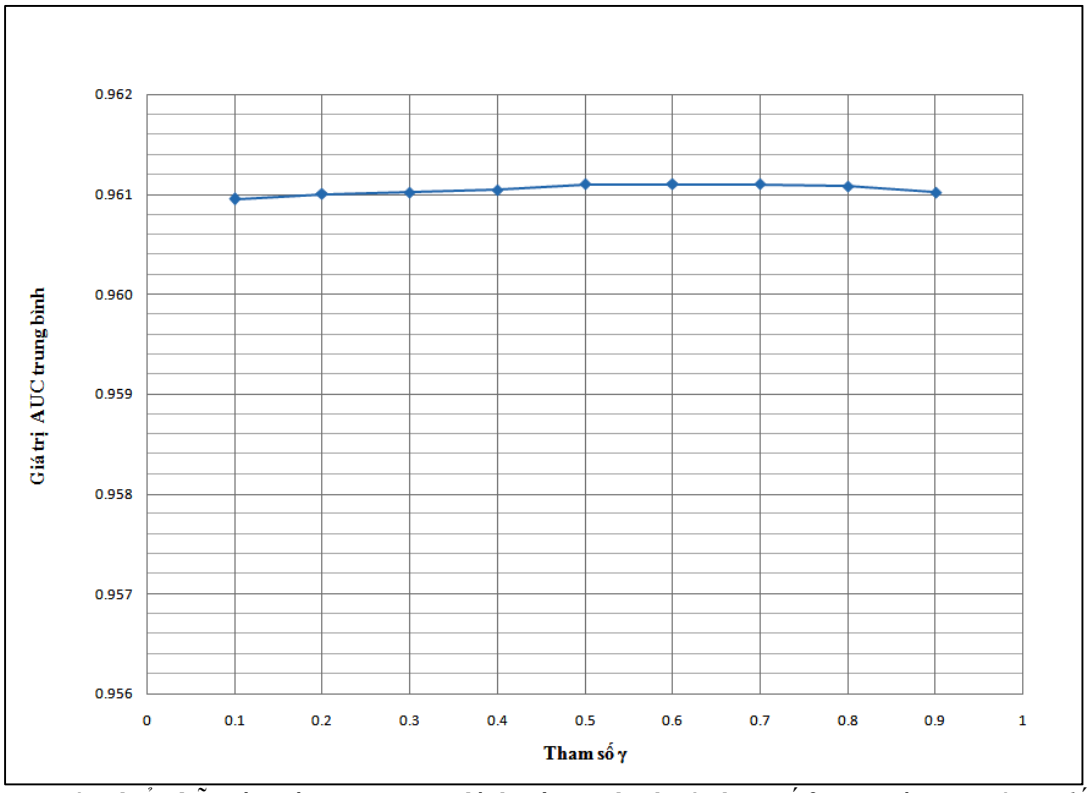
trong đó: *TP* (true positive) là số trường hợp thử nghiệm mà thứ hạng của $s \leq \tau$, *FN* (false negative) là số trường hợp thử nghiệm mà thứ hạng của $s > \tau$, *FP* (false positive) là số trường hợp thử nghiệm mà thứ hạng của $c \leq \tau$ (với mỗi $c \in C$), và *TN* (true negative) là số trường hợp thử nghiệm mà thứ hạng của $c > \tau$ (với mỗi $c \in C$). Một cặp giá trị *sensitivity* và *1-specificity* tương ứng với một điểm trên đường cong ROC.

## III. KẾT QUẢ THỰC NGHIỆM

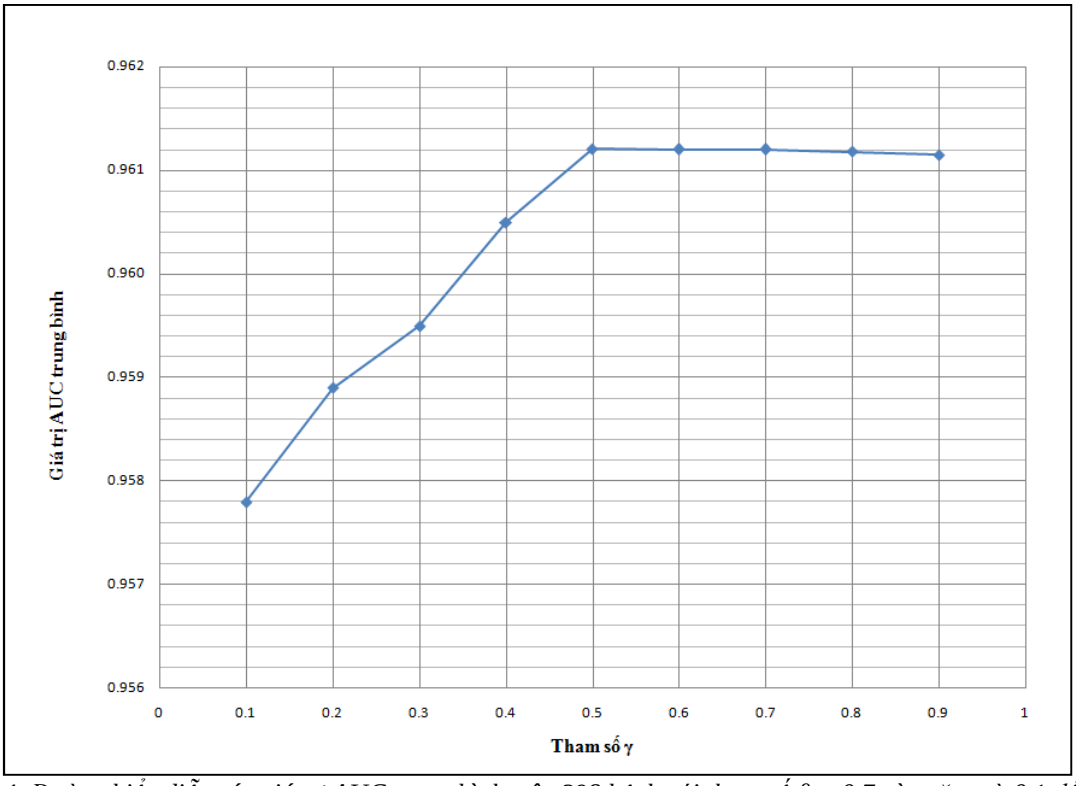
Trong phần này, chúng tôi đánh giá ảnh hưởng của các tham số tới tính ổn định của thuật toán đồng thời lựa chọn bộ tham số cho độ chính xác tốt nhất. Sau đó, chúng tôi so sánh độ chính xác của phương pháp đề xuất với PageRank with priors trên cùng bộ dữ liệu theo giá trị AUC.

### III.1. Ảnh hưởng của các tham số

Thực nghiệm đầu tiên được tiến hành để đánh giá ảnh hưởng của các tham số tới hiệu quả của phương pháp dựa trên dữ liệu và phương pháp đánh giá đã nêu ở Phần 2. Cụ thể, đối với mỗi bệnh, chúng tôi đã tiến hành phân hạng các gen ứng viên và đánh giá hiệu quả phân hạng thông qua giá trị AUC. Giá trị AUC trung bình trên 398 bệnh được sử dụng làm kết quả để đánh giá độ chính xác của phương pháp.




*Hình 3. Đường biểu diễn các giá trị AUC trung bình trên 398 bệnh với tham số β = 0.8 và γ tăng từ 0.1 đến 0.9*



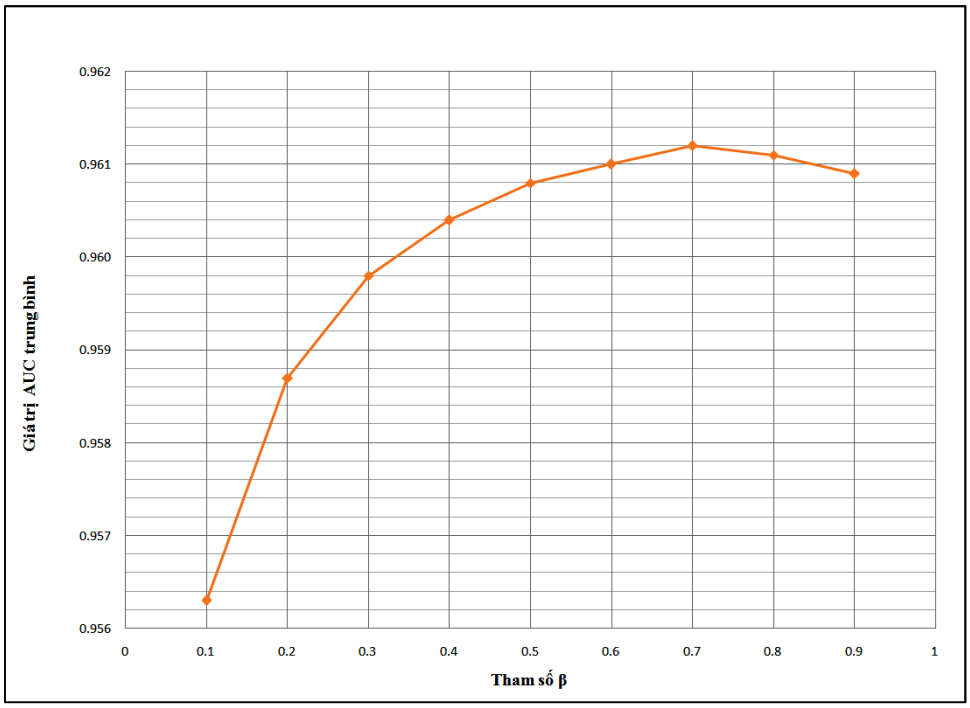

*Hình 4. Đường biểu diễn các giá trị AUC trung bình trên 398 bệnh với tham số β = 0.7 và γ tăng từ 0.1 đến 0.9*

Dựa trên kết quả thử nghiệm sơ bộ, chúng tôi thấy rằng: với giá trị $\beta$ >= 0.8, khi tăng hay giảm giá trị $\gamma$, kết quả thực hiện thuật toán hầu như không thay đổi. Hình 3 cho thấy với $\beta$ = 0.8, giá trị AUC trung bình ổn định khi $\gamma$ biến thiên trong khoảng [0,1]. Điều này là do khi xác suất quay trở lại gốc lớn, các nút gần các nút gốc được thăm nhiều hơn, trong khi các nút ở xa các nút gốc ít được thăm hơn, do đó giá trị điểm thưởng không thay đổi nhiều, dẫn tới kết quả phân hạng chung ít thay đổi. Trong trường hợp $\beta$ <= 0.7 và khi $\gamma$ tăng từ 0.1 đến 0.4, giá trị điểm thưởng tăng dần, dẫn đến thứ hạng của gen thử nghiệm cũng tăng theo và thứ hạng này ổn định khi $\gamma$ >= 0.5. Hình 4 biểu diễn kết quả thử nghiệm với $\beta$ = 0.7 và $\gamma$ biến thiên trong khoảng [0,1]. Với các trường hợp $\beta$ = [0.1, 0.2, 0.3, 0.4, 0.5, 0.6], chúng ta cũng thu được kết quả tương tự.

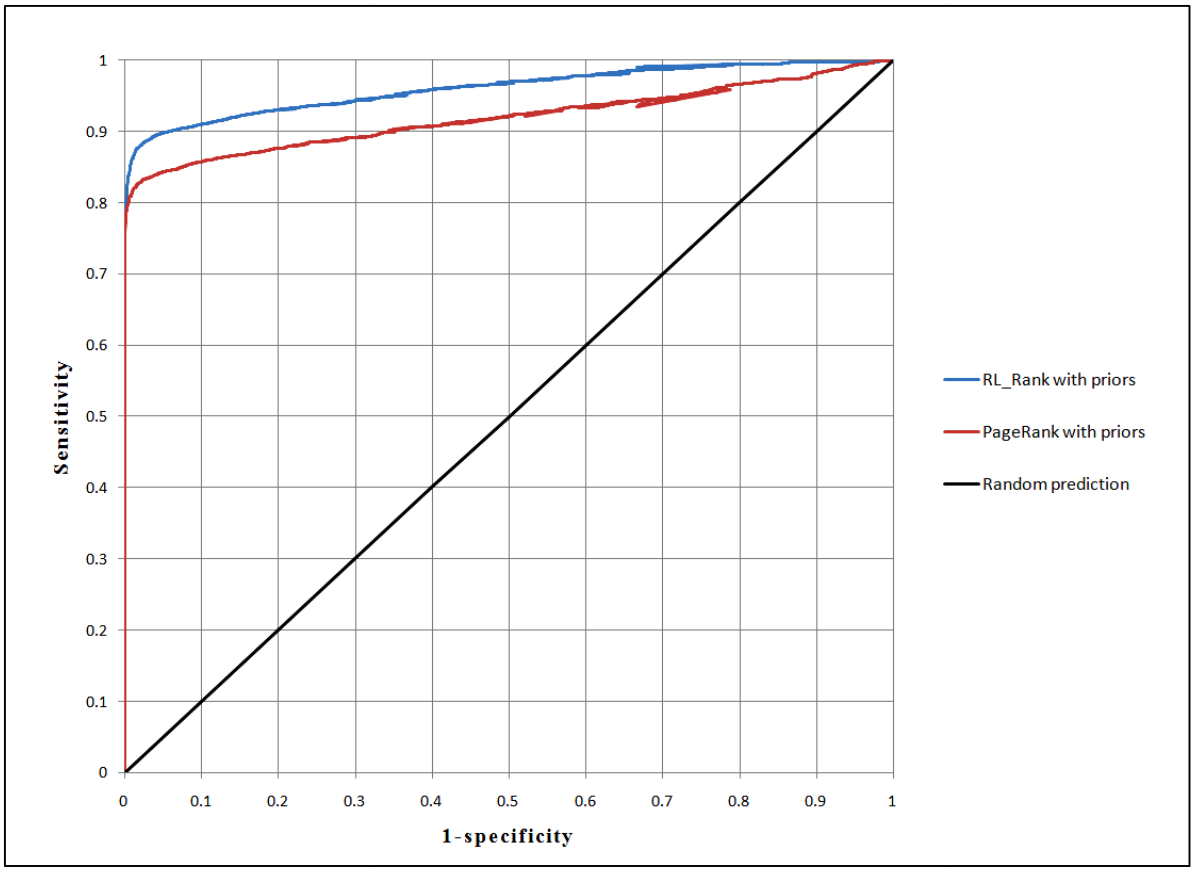
Tham số $\beta$ là xác suất quay lại trong thuật toán PageRank with priors. Để xác định ảnh hưởng của tham số $\beta$ tới hiệu quả của phương pháp đề xuất, chúng tôi thiết lập $\gamma$ = 0.5 và tính giá trị AUC trung bình trên 398 bệnh cho mỗi giá trị $\beta$ khi $\beta$ tăng từ 0.1 đến 0.9. Kết quả thực nghiệm được thể hiện trong Hình 5 cho thấy độ chính xác của thuật toán (thể hiện qua giá trị AUC trung bình) không thay đổi nhiều khi thay đổi $\beta$. Cụ thể, giá trị cao nhất đạt được khi $\beta$ = 0.7 chỉ chênh lệch khoảng 1% so với giá trị thấp nhất khi $\beta$ = 0.1.

### III.2. So sánh với PageRank with priors và các thuật toán cùng lớp

Trong thực nghiệm tiếp theo, chúng tôi tiến hành so sánh kết quả phân hạng của phương pháp đề xuất với PageRank with priors trên cùng một bộ dữ liệu và phương pháp đánh giá LOOCV đã được giới thiệu. Chúng tôi thiết lập các tham số tương ứng $\gamma$ = 0.5 và $\beta$ = 0.7 (đây là bộ tham số cho kết quả phân hạng tốt nhất theo đánh giá ở Phần III.1). Đối với mỗi phương pháp, chúng tôi tiến hành vẽ đường cong ROC và tính giá trị AUC trung bình cho tất cả 398 bệnh bằng cách tính các giá trị *sensitivity* và *1-specificity* cho từng bệnh, sau đó tính giá trị *sensitivity* và *1-specificity* trung bình của 398 bệnh tại các ngưỡng $\tau$. Hình 6 thể hiện đường cong ROC của cả hai phương pháp được so sánh.




*Hình 5. Đường biểu diễn các giá trị AUC trung bình trên 398 bệnh với tham số β tăng từ 0.1 đến 0.9*




*Hình 6. Đường cong ROC biểu diễn kết quả thuật toán RL_Rank with priors với các tham số γ = 0.5, β = 0.7 và PageRank with priors với tham số β = 0.7*

*Bảng 1. Kết quả của RL_Rank with priors và các thuật toán cùng lớp*

| **Phương pháp** | **Tham số β** | **Giá trị AUC trung bình** |
|---|---|---|
| RL_Rank with priors | β = 0.7 | 0.9612 |
| PageRank with priors | β = 0.7 | 0.9192 |
| Random Walk with Restart | β = 0.7 | 0.9364 |
| K-Step Markov | K = 6 | 0.9087 |

Để khẳng định thêm về hiệu quả của phương pháp đề xuất so với các phương pháp cùng lớp, chúng tôi cũng đã tiến hành thực nghiệm phương pháp RWR (Random Walk with Restart) và K-Step Markov trên cùng bộ dữ liệu và phương pháp đánh giá. Kết quả thực nghiệm và giá trị các tham số thiết lập được trình bày trong Bảng 1.

Dựa trên các kết quả thực nghiệm thu được, chúng tôi nhận thấy rằng so với các phương pháp phân hạng dựa trên mạng tương tác protein khác như RWR, K-step Markov thì kết quả phân hạng của phương pháp chúng tôi đề xuất cũng đạt hiệu quả cao hơn. Điều này cho thấy độ chính xác của thuật toán RL_Rank with priors tốt hơn là bởi sự kết hợp yếu tố tăng cường trong kết quả phân hạng.

### III.3. Kiểm chứng về mặt sinh học đối với bệnh cao huyết áp

Trong phần này, nhằm mục đích kiểm chứng về mặt sinh học cho phương pháp đề xuất, chúng tôi tiến

hành phân hạng các gen liên quan đến bệnh cao huyết áp (hypertension) có mã OMIM 145500 và thu thập các bằng chứng y văn của các gen có thứ hạng cao trong kết quả phân hạng. Đối với bệnh cao huyết áp, chúng tôi tra cứu trong cơ sở dữ liệu bệnh - gen và xác định được 18 gen bệnh tương ứng (xem Bảng 2a).

*Bảng 2a. Danh sách các gen gây bệnh cao huyết áp đã biết và số liên kết trong mạng tương tác protein*

| Stt | Gen ID | Gen | Số liên kết |
|---|---|---|---|
| 1 | 6401 | SELE | 12 |
| 2 | 1577 | CYP3A5 | 55 |
| 3 | 8490 | RGS5 | 2 |
| 4 | 50986 | HYT2 | 0 |
| 5 | 444980 | HYT4 | 0 |
| 6 | 1889 | ECE1 | 13 |
| 7 | 5740 | PTGIS | 10 |
| 8 | 118 | ADD1 | 2 |
| 9 | 100188807 | HYT5 | 0 |
| 10 | 100188808 | HYT6 | 0 |
| 11 | 183 | AGT | 95 |
| 12 | 185 | AGTR1 | 20 |
| 13 | 117191 | HYT1 | 0 |
| 14 | 2784 | GNB3 | 44 |
| 15 | 481 | ATP1B1 | 8 |
| 16 | 4843 | NOS2 | 36 |
| 17 | 4846 | NOS3 | 36 |
| 18 | 387575 | HYT3 | 0 |
| **Tổng số liên kết** | | | **333** |

Tiếp theo, chúng tôi tiến hành phân tích mạng tương tác protein và trích rút được 333 liên kết với 12 gen bệnh đã biết (6 gen còn lại không có liên kết trong mạng tương tác protein). Các gen này được sử dụng như tập huấn luyện để tiến hành phân hạng.

Sau khi tiến hành phân hạng, chúng tôi lựa chọn 20 gen có thứ hạng cao nhất (xem Bảng 2b) và tiến hành thu thập các bằng chứng về mối quan hệ giữa các gen này và bệnh cao huyết áp từ các y văn được công bố trong cơ sở dữ liệu PubMed, OMIM, GeneRIFs và BioSystems.

Từ kết quả tra cứu thu thập được, chúng tôi thấy rằng có 9 gen đã được báo cáo liên quan trực tiếp gây ra bệnh cao huyết áp (các gen đánh dấu * trong Bảng 2b). Các gen có thứ hạng cao còn lại mặc dù không có bằng chứng trực tiếp liên quan đến bệnh cao huyết áp nhưng chúng lại có liên quan đến các bệnh có thể là nguyên nhân gây ra bệnh cao huyết áp như rối loạn chuyển hóa kẽm hoặc tiểu đường. Đối với các gen này, chúng tôi xem là những đề xuất cho các nhà y - sinh học nghiên cứu, tìm kiếm thêm bằng chứng liên quan trực tiếp tới bệnh cao huyết áp trong tương lai.

*Bảng 2b. Danh sách 20 gen có thứ hạng cao theo kết quả phân hạng của RL-Rank with prior*

| Hạng | Gen ID | Gen |
|---|---|---|
| 1 | 1906 | EDN1* |
| 2 | 2641 | GCG* |
| 3 | 2740 | GLP1R* |
| 4 | 6708 | SPTA1 |
| 5 | 119 | ADD2* |
| 6 | 4691 | NCL |
| 7 | 4670 | HNRNPM |
| 8 | 4878 | NPPA* |
| 9 | 1584 | CYP11B1* |
| 10 | 1589 | CYP21A2* |
| 11 | 4881 | NPR1* |
| 12 | 3175 | ONECUT1 |
| 13 | 7781 | SLC30A3 |
| 14 | 7779 | SLC30A1 |
| 15 | 3670 | ISL1 |
| 16 | 6517 | SLC2A4* |
| 17 | 148867 | SLC30A7 |
| 18 | 55532 | SLC30A10 |
| 19 | 1588 | CYP19A1 |
| 20 | 3643 | INSR |

## IV. KẾT LUẬN

Trong bài báo này, chúng tôi đã đề xuất được phương pháp kết hợp khái niệm học tăng cường với thuật toán PageRank with priors và áp dụng cho bài toán phân hạng gen nhằm mục đích tìm kiếm các gen gây bệnh dựa trên mạng tương tác protein và mối quan hệ bệnh-gen đã biết. Kết quả thực nghiệm cho thấy phương pháp đề xuất có thể đạt được kết quả tốt hơn so với phương pháp trước đó sử dụng thuật toán PageRank with priors. Các kết quả cũng cho thấy việc các mạng sinh học có những điểm tương đồng với các mạng xã hội và mạng Web, đã cho phép ứng dụng và mở rộng các thuật toán phân hạng trang Web cho bài toán phân hạng và tìm kiếm gen gây bệnh. Với kết quả này, chúng ta có thể thấy rằng khả năng dự đoán gen bệnh mới dựa trên độ quan trọng tương đối/độ

tương tự của chúng so với các gen bệnh đã biết là hoàn toàn khả thi và các gen ứng viên có thứ hạng cao có thể sử dụng làm gợi ý ban đầu cho các nhà sinh học kiểm tra bằng các thí nghiệm sinh học chuyên sâu.

Phương pháp đề xuất trong bài báo có thể được phát triển thành một phần mềm ứng dụng để triển khai trong các cơ sở nghiên cứu về y sinh học phục vụ công tác nghiên cứu và đào tạo. Đồng thời cũng có thể ứng dụng để giải quyết bài toán phát hiện các gen liên quan đến những căn bệnh cụ thể như một số bệnh ung thư, tiểu đường, alzheimer .... Đó cũng là bước tiền đề cho việc tìm ra các phương pháp điều trị thích hợp cho các bệnh liên quan đến gen.

Trong các nghiên cứu tiếp theo, chúng tôi sẽ thực hiện phương pháp kiểm nghiệm kết quả bằng cách tìm các bằng chứng y sinh học trên các tài liệu y văn về mối liên quan giữa các gen ứng viên có thứ hạng cao và bệnh đang được xem xét. Đồng thời chúng tôi sẽ thử nghiệm với các thuật toán tính độ trung tâm hoặc tầm quan trọng khác của các nút trên đồ thị và áp dụng cho bài toán tìm/phân hạng gen ứng viên gây bệnh. Thêm vào đó, chúng tôi cũng sẽ ứng dụng thuật toán đề xuất dựa trên các mạng sinh học khác như: mạng trao đổi chất, mạng điều hòa gen, mạng tương tác di truyền.

## LỜI CẢM ƠN

Nghiên cứu này được tài trợ bởi Quỹ phát triển khoa học và công nghệ quốc gia (NAFOSTED) trong đề tài mã số 102.01-2014.21

## TÀI LIỆU THAM KHẢO

[1]. PEREZ-IRATXETA, C., BORK, P. and ANDRADE, M.A, "*Association of genes to genetically inherited diseases using data mining*", Nat Genet, vol. 31, pp. 316-319, 2002.

[2] CARLOS ROBERTO ARIAS, HSIANG-YUAN YEH,VON-WUN SOO, "*Disease Gene Prioritization*", Selected Works in Bioinformatics, Dr. Xuhua Xia (Ed.), ISBN: 978-953-307-281-4, InTech, Available from *http://www.intechopen.com/books/selected-works-in-bioinformatics/disease-gene-prioritization*, 2011.

[3] XIUJUAN WANG, NATALI GULBAHCE and HAIYUANYU, "*Network-based methods for human disease gene prediction*", Briefings in Functional Genomics, vol. 10, no. 5, pp. 280-239, 2011.

[4] D. J. WATTS, S. H. STROGATZ, "*Collective dynamics of small-world networks*", Nature 393(1), pp. 440-442, 1998.

[5] JUNKER BH, KOSCHUTZKI D, SCHREIBER F, "*Exploration of biological network centralities with CentiBiN*", BMC Bioinformatics, p. 219, July 2006.

[6] LOVASZ, L., "*Random walks on graphs: A survey*", Combinatorics, Paul Erdos is Eighty, vol. 2, pp. 353-398, 1996.

[7] KOHLER, S., BAUER, S., HORN, D. and ROBINSON, P., "*Walking the Interactome for Prioritization of Candidate Disease Genes*", The American Journal of Human Genetics, vol. 82, pp. 949-958, 2008.

[8] DUC-HAU LE, YUNG-KEUN KWON, "*Neighbor-favoring weight reinforcement to improve random walk-based disease gene prioritization*", Computational Biology and Chemistry, vol. 44, pp. 1–8, 2013.

[9] DUC-HAU LE, YUNG-KEUN KWON, "*GPEC: A Cytoscape plug-in for random walk-based gene prioritization and biomedical evidence collection*", Computational Biology and Chemistry, vol. 37, pp. 17–23, 2012.

[10] CHEN, J., ARONOW, B. and JEGGA, A, "*Disease candidate gene identification and prioritization using protein interaction networks*", BMC Bioinformatics, vol. 10, p. 73, 2009.

[11] VANUNU O, MAGGER O, RUPPIN E, et al., "*Associating genes and protein complexes with disease via network propagation*", PLoSComput Biol, vol. 6:e1000641, 2010.

[12] VALI DERHAMI, ELAHE KHODADADIAN, MOHAMMAD GHASEMZADEH, ALI MOHAMMAD ZAREH BIDOKI, "*Applying reinforcement learning for web pages ranking algorithms*", Applied Soft Computing, vol. 13, pp. 1686–1692, 2013.

[13] ELAHE KHODADADIAN, MOHAMMAD GHASEMZADEH, VALI DERHAMI, S.ALI MIRSOLEIMANI, "*A Novel Ranking Algorithm Based on Reinforcement Learning*", Artificial Intelligence and

Signal Processing (AISP), 2012 16th CSI International Symposium on, pp. 546-551, 2012.

[14] HAVELIWALA, T., "*Topic-sensitive PageRank*", Proceedings of the 11th International World Wide Web Conference, Honolulu, Hawaii, pp. 517-526, 2002.

[15] JEH, G. and WIDOM J., "*Scaling personalized Web search*", Stanford University, Computer Science Department Technical Report, 2002.

**Nhận bài ngày**: 11/8/2014

## SƠ LƯỢC VỀ TÁC GIẢ

**ĐẶNG VŨ TÙNG**

Sinh năm 1972.

Tốt nghiệp ĐH Tổng hợp Hà Nội năm 1995; Thạc sỹ chuyên ngành Hệ thống thông tin năm 2011, NCS khóa 2013 tại Học viện Công nghệ bưu chính viễn thông.

Hiện công tác tại bộ môn Tin học, Học viện Thanh thiếu niên Việt Nam.

Lĩnh vực nghiên cứu: hệ thống thông tin, tin sinh học.

Điện thoại: 0913542479

Email: tung_dv@yahoo.com

**DƯƠNG ANH TRÀ**

Sinh năm 1981.

Tốt nghiệp ĐH Bách khoa Hà Nội năm 2004; Thạc sỹ chuyên ngành Hệ thống thông tin tại Học viện Kỹ thuật quân sự năm 2014.

Hiện công tác tại Viện Nghiên cứu và Phát triển Viettel.

Lĩnh vực nghiên cứu: Hệ thống thông tin, các hệ thống chỉ huy và điều khiển.

Điện thoại: 0985066505

Email: duonganhtra@gmail.com

**LÊ ĐỨC HẬU**

Sinh năm 1979

Tốt nghiệp ĐH Bách khoa Hà Nội năm 2002

Thạc sỹ khoa học ĐH Bách Khoa Hà nội năm 2008

Bảo vệ tiến sĩ năm 2012 tại Đại học Ulsan, Hàn Quốc.

Hiện công tác tại Trung tâm Tin học, Đại học Thủy Lợi.

Lĩnh vực nghiên cứu: học máy và khai phá dữ liệu, tin sinh học và ứng dụng, phân tích và khai phá mạng xã hội, lập trình song song trên GPU với CUDA và OpenCL.

Điện thoại: 0912324564

Email: hauldhut@gmail.com

**TỪ MINH PHƯƠNG**

Sinh năm 1971

Tốt nghiệp ĐH Bách khoa Taskent, Uzobekistan.

Bảo vệ tiến sĩ năm 1995 tại Viện Điều khiển học thuộc Viện hàn lâm khoa học Uzobekistan.

Hiện công tác tại Khoa CNTT, Học viện Công nghệ Bưu chính viễn thông.

Lĩnh vực nghiên cứu: ứng dụng của học máy, tin sinh học.

Điện thoại: 0913507508

Email: phuongtm@ptit.edu.vn